Drawings

Applicant will submit formal drawings when the application is allowed.

35 USC 103

Claims 1 and 31-39 have been rejected under 35 U.S.C. 103(a) over Gupta et al. (US Patent 5,535,116) in view of Hagersten et al. (US Patent 5,860,109).

Applicant traverses.

Each of applicant's claims recite (1) a shared memory where sections of the same global memory are located at different nodes, (2) a system of cache coherency over a multi-stage network (not a bus), and (3) a cache coherency where shared memory always contains the latest copy of changed data.

With respect to Gupta, as the Examiner states in the Office Action, "Gupta fails to specifically teach the features... each processing node including a unique section of shared memory which is not a cache and an adapter for

storing changed data to each unique section so that each section contains the most recent data." However, the Examiner states that Hagersten teaches these features.

With respect to Hagersten, as applicant will explain hereafter, Hagersten teaches none of these features.

Applicant asserts that neither Hagersten or Gupta, nor their combination, teaches the claimed features described above.

First, Hagersten teaches that all coherency functions are performed in regards to a single memory 110 which is a totally independent memory and has its own local physical addresses (Hagersten Col. 3, lines 14-15). This means that Hagersten's system has no global address system where sections of memory stored at each node contribute a portion of global memory to the overall distributed memory. Instead, Hagersten's memory is an individual and totally independent memory having its own local physical address scheme. Each of applicant's claims distinguishes Hagersten by reciting a shared memory where sections of the same global memory are located at different nodes.

Second, Hagersten teaches a bus-based coherency system

with bus 108 in Figure 2 interconnecting processor groups 102, 104, and 106 to central memory 110 and coherence transformer 220. Thus, Hagersten confines all cache coherency to a single bus in the system - bus 108 of Figure 2.

Applicant, on the other hand, uses a significantly different architecture comprising a plurality of single processors interconnected by a high speed network and performs cache coherency throughout the entire system. Figures 3 and 4 of the present invention show how the nodes of the system are interconnected via a network using switches, not busses. In applicant's system, "a plurality of nodes communicate via messages over an interconnection network..." (Page 1, lines 15-16).

In contrast, Hagersten teaches a cache coherency system for a bus-based computer system which requires a coherence transformer 220 enabling several external devices to perform accesses from bus-based memory 110 and keep cache coherency (Hagersten, Col. 6, line 66 to Col. 7, line 3). Hagersten teaches the bus operation as follows:

"...when a memory access request is issued by one of the bus entities on common bus 108... this memory

access request is forwarded to all bus entities" simultaneously. (Col. 11, lines 9-14.)

This is characteristic of all bus coherency systems, where all entities on the bus can monitor all bus activity as the bus passes all signals to all entities on the bus. Cache coherency on a bus is quite simple because every bus entity monitors (snoops) every access of memory. This is not the case in a network coherency system, which is more difficult because no node on the network is aware of the internal memory accesses to the memories at other nodes, and there is no bus to snoop. The present invention provides for maintaining cache coherency over a network including an unlimited number of parallel processors accessing a global distributed memory. Applicant's claims recite a system of cache coherency over a multi-stage network (which is not a bus), and thus distinguish Hagersten.

Third, Hagersten teaches a different method of cache coherency than that which is recited in applicant's claims. Conventionally, coherency methods have been classified as either "write-back" or "write-thru". For the write-thru method, changed data is immediately stored to shared memory, so that the most recent data is always resident in the shared memory. Hagersten teaches a write-back method where

changed data is held in a cache in system 100 of Figure 2 or at an external device connected to bus 108 via coherence transformer 220. Changed locations might never be written to memory 110. Hagersten explains that "the external device owning the exclusive copy can unilaterally modify this copy without having to inform other bus entities attached to common bus 108." (Hagersten, Col. 10, lines 64-67).

In contrast, applicant's invention relates to a write-thru method of cache coherency. This is explained as follows: "...a method for maintaining cache coherency efficiently across a network using distributed directories, invalidation, read request, and write-thru updates." (Page 1, lines 22-24). This write-thru feature distinguishes Hagersten and is recited in all of applicant's claims with statements such as "changing said shared memory, wherein changed data is stored immediately to shared memory regardless of which of said noes is changing the data and which of said nodes includes the section of shared memory to be changed, wherein said shared memory always contains the most recent data".

Further with respect to the second distinction, Hagersten does teach communication with entities outside of the bus-based system 100, but the communication is not via a

network and no cache coherency is performed externally but only on internal bus 108. In connection with Figure 2, Hagersten teaches "coherence transformer 200 may communicate with... any of external devices 202, 204, and 206 using a protocol that is appropriate for the external device with which it communicates." In other words, the external devices are not interconnected by a homogeneous network, but instead are interconnected by specialized point-to-point links native to each individual device.

In conclusion, neither Hagersten nor Gupta, nor their combination, teach the above three aspects recited in each of applicant's claims: (1) a shared memory where sections of the same global memory are located at different nodes, (2) a system of cache coherency over a multi-stage network (not a bus), and (3) a cache coherency where shared memory always contains the latest copy of changed data.

SUMMARY AND CONCLUSION

Applicant urges that the case be passed to issue with claims 1 and 31-39.

If, in the opinion of the Examiner, a telephone

conversation with applicant(s) attorney could possibly facilitate prosecution of the case, he may be reached at the number noted below.

Sincerely,

Howard T. Olnowich

By

Shelley M Beckstrand
Reg. No. 24,886

Date: 20 Aug 2001

Shelley M Beckstrand, P.C.
Attorney at Law
314 Main Street
Owego, NY 13827

Phone: (607) 687-9913
Fax: (607) 687-7848